

On Complementary and Independent Mappings on Databases

Arthur M. Keller and Jeffrey D. Ullman
Stanford University

ABSTRACT. We define the notion of independent views to indicate whether the range values of the two views may be achieved independently. The concept of complementary views indicates when the domain element can be uniquely determined by the range values of the two complementary views. We consider the relationship between independent and complementary views. In unrestricted domains, a view (but not the identity or empty view) can have more than one complementary, independent view. Databases, however, are more restricted domains: They are finite power sets. A view is monotonic if it preserves inclusion. However, in finite power sets when all views are monotonic, if a given view has another view which is independent and complementary, then this view is unique.

KEYWORDS. Relational databases, database theory, complementary mappings, independent mappings, view update.

CR CATEGORIES. H.2.1, H.1.1, E.4.

Introduction

The problem of updating databases through views is an important practical problem that has attracted much theoretical interest [Bancilhon 79, 81, Carlson 79, Davidson 81, Dayal 78, 79, 82, Furtado 79, Kaplan 81, Keller 82]. A database is a subset of a finite power set, and a database view is a (total, many-to-one) mapping from one finite power set to another. The user specifies queries to be executed against the database view; these queries are translated to queries against the underlying

database through query modification [Stonebraker 75]. However, in current practice, updates must be specified against the underlying database rather than against the view.

Since the view is only an uninstantiated window onto the database, any updates specified against the database view must be translated into updates against the underlying database. The updated database state then induces a new view state, and it is desired that the new view state corresponds to performing the user-specified update directly on the original view state, were that possible. This is described by the following diagram.

$$\begin{array}{ccc} V(DB) & \xrightarrow{U} & V(DB') \\ \uparrow V & & \uparrow V \\ DB & \xrightarrow{T(U)} & DB' \end{array}$$

The user specifies update U against the view of the database, $V(DB)$. The view update translator T supplies the database update $T(U)$, which results in DB' when applied to the database. The new view state is $V(DB')$. This translation has *no side effects in the view* if $V(DB') = U(V(DB))$, that is, if the view has changed precisely in accordance with the user's request. In this paper, we require that all view update translators have no side effects in the view.

Given a view definition, the question of choosing a view update translator arises. This requires understanding the ways in which individual view update requests may be satisfied by database updates. Any particular view update request may result in a view state that does not correspond to any database state. Such a view update request may not be translated without relaxing the constraint precluding view side effects.* Otherwise, the update request is rejected by the view update translator. If we are lucky, there will be precisely one way to perform the database update that results in the desired view update. Since the view is many-to-one, the new view state may correspond to many database states. Of these database states, we would like to choose one that is "as close as possible" under some measure to the original database state. That is, we would like to minimize the effect of the view update on the database.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association of Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

This work was supported in part by contract N00039-82-G-0250 (the Knowledge Base Management Systems Project, Prof. Gio Wiederhold, Principal Investigator) from the Defense Advanced Research Projects Agency and by contract AFOSR-80-0212 (Universal Relations, Prof. Jeff Ullman, Principal Investigator) from the Air Force Office of Scientific Research, both of the United States Department of Defense. The views and conclusions contained in this document are those of the authors and should not be interpreted as representative of the official policies of DARPA or the US Government.

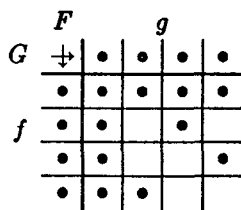
Authors' address: Computer Science Department, Stanford University, Stanford, CA 94305-2085.

* In certain cases, we have shown that it is quite reasonable to relax this constraint in a limited manner [Keller 82].

One way to express a limitation of effects of view updates on the database is through the concept of *constant complements* [Bancilhon 81]. Two views are complementary if given the state of each view, there is a unique corresponding database state. Intuitively, this means that the two views are sufficient to reconstruct the database. Bancilhon and Spyrtos have observed that by choosing a complementary view and holding it constant, that there is at most one way to translate any update on the given view. They have also shown that if a view is not empty or the identity, then it has multiple minimal complements. (A complement is *minimal* if no view providing less information is also complements. Providing more information does not adversely affect complementarity; therefore, the issue is only interesting when we consider minimal complements.)

We observe that choosing a constant complement may cause the view update translator to reject requests that have translations (although none of those translations keep the complement constant). We define two views as independent when any pair of view states corresponds to a database state. When independent views are complementary, it is always possible to hold the state of one view constant while generating any possible state of the other view. Thus, choosing an independent complement (if one exists) permits all updates expressed against the view to be translated to updates expressed against the database. The question then arises whether a view has multiple independent complements. To answer this question, we define a view as monotonic if it preserves inclusion (recall that the domain and range of a view is a finite power set). Informally, a view is monotonic if adding tuples to the database does not remove any tuples from the view (although it could augment the view). There are non-monotonic views that have multiple independent complements. However, a monotonic view has at most one complement that is independent and monotonic.

The following diagram illustrates two database mappings f and g .

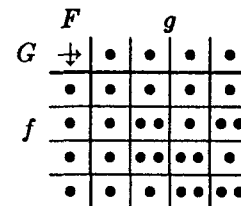


Each • represents one database state. The domain D is represented by the •'s located in the main quadrant of the diagram (the lower right from the heavy lines). The function f maps database states of D into database

states in F (the left column) by moving across the row. Similarly, the function g maps database states of D into database states in G (the top row) by moving up along the column. The equivalence classes of D induced by f (D/f) are the rows of D . And the equivalence classes of D induced by g (D/g) are the columns of D . We can take the intersection of an equivalence class of D/f and one from D/g ; this is represented by one box of the diagram. The two mappings f and g are complementary if given a database state in F and one in G , there is at most one database state in D that maps into both database states (by their respective mappings). That means each box of the diagram has at most one • in it. This particular diagram illustrates two complementary functions.

Updating the database D through the user view f involves changing view states from some database in F to another (in F). If the mapping g is to be held constant, then the new database state is found by moving along the same column (for the same image in G) from the original row to the desired row. If the mapping to be held constant g is complementary to f , then there is at most one candidate resultant database in D that maps to the same database in G (by g) and maps to the new database in F (by f). When the box (intersection of the view and complement equivalence classes) is empty, the view update request cannot be performed exactly while preserving the complement—the request is rejected; when the intersection contains exactly one database, the view update request has a unique translation that preserves the complement.

The following diagram illustrates two independent functions. Each box in D in the diagram contains at least one •.



Again, updating the database D through the user view f involves changing view states from some database in F to another (in F). If the mapping g is to be held constant, then the new database state is found by moving along the same column (for the same image in G) from the original row to the desired row. If the mapping to be held constant g is complementary to f , then there is at least one candidate resultant database in D that maps to the same database in G (by g) and maps to the new database in F (by f). When the box (intersection of the view and complement equivalence classes)

contains exactly one database, the view update request has a unique translation that preserves the complement; when the intersection contains more than one database, the view update request has several candidate translations that preserve the complement—the request is ambiguous.

Holding a view complement constant means that whenever the view update is translatable, that translation is unique. If the view held constant is independent (from the user view), then all view update requests can be translated, perhaps ambiguously. If the view that is held constant is both complementary to and independent of the user view, then all view update requests are unambiguously translatable into database update requests. When the views are constrained to be monotonic, if a view has a independent complement, it is unique.

We will now proceed to a formal treatment of the results we have stated informally.

Definitions

It is assumed that the reader is familiar with database theory [Ullman 83] and set theory [Halmos 60].

A database is a finite power set.

DEFINITION. Let f and g be two functions whose domain is D . (Here we are not concerned with the range of f and g , but only with the equivalence classes induced on D by f and g .) Then f and g are *independent mappings* if

$$\begin{aligned} & [\forall x, y][((\exists d_1 \in D)(f(d_1) = x)) \wedge \\ & ((\exists d_2 \in D)(g(d_2) = y)) \rightarrow \\ & ((\exists d \in D)(f(d) = x \wedge g(d) = y))]. \end{aligned}$$

The notion of independence we use here is different from Rissanen's notion of independence [Rissanen 77]. His notion stated that two components were independent when the original database could be obtained from them by lossless joins that preserved all dependencies. Our definition relates to the ability to change the selected range value of one mapping while keeping the range value of the other mapping constant. This definition is useful for the problem of view updates, where it is important to consider whether an update specified through a view may be done without affecting another view.

DEFINITION [Bancilhon 81]. Let f and g be two functions whose domain is D . Then f and g are *complementary mappings* if

$$[\forall x, y \in D][(x \neq y) \wedge f(x) = f(y) \rightarrow g(x) \neq g(y)].$$

DEFINITION. Two functions (mappings) f and g with the domain D are *equivalent* if they induce the same equivalence class on D . (That is $D/f = D/g$. Recall that D/f is defined as follows: $\forall d \in D, \forall d' \in D, d$ and d' are in the same member of D/f iff $f(d) = f(d')$.)

We observe that independence and complementarity are different properties. Independence means that function can generate all values of its range while another function has a specific range value. Complementarity means that there is at most one element of the domain that simultaneously results in any pair of range values, one from each of two functions. We can give another characterization of these two concepts. Each function f generates a set of equivalence classes D/f . Given two functions f and g we can take the intersection of equivalence classes of D/f with equivalence classes of D/g . If all of these intersections have at most one (domain) element in them, the two functions are complementary. If all of these intersections have at least one (domain) element in them, the two functions are independent.

DEFINITION. Let f and g be complementary and independent functions whose domain is D , and let h be an arbitrary function whose domain is also D . Let the range of f be F and the range of g is G . Since f and g are complementary and independent, there is a one-to-one correspondence between $F \times G$ and D ; that is, $d \in D$ corresponds to $a \times b$ where $a = f(d)$ and $b = g(d)$. Then the *coordinatization* of h over f and g is the function h' whose domain is $F \times G$ such that $h(d) = h'(f(d), g(d))$. (We note that h is equivalent to some h_0 iff h' is.)

One question is when is there a unique (up to equivalence) complementary, independent function g for a function f . For example, let $f(x, y) = x$ and $g(x, y) = y$. It is clear that these are independent and complementary. The function $g'(x, y) = 2y$ is independent and complementary to f but also equivalent to g . However, the function $g''(x, y) = x + y$ is independent and complementary to f but not equivalent to g . Since our domain of interest is relational databases, and the mappings of interest are relational views consisting of combinations of select, project, join, and union, we will use a property of these mappings.

Monotonic Functions

DEFINITION. An n -ary function f whose domain is a finite power set is *monotonic* if $(\forall i)(R_i \subseteq S_i) \rightarrow f(R_1, \dots, R_n) \subseteq f(S_1, \dots, S_n)$. (Select, project, join, and union of relations are monotonic functions. The

set difference operator, however, is not monotonic.)*

THEOREM. The composition of monotonic functions is monotonic.

PROOF. Let f and g be monotonic functions (possibly with multiple arguments). Let h be some composition of f and g . There are two issues to handle. The first is that the function g is used as one or more arguments of f to form h . The second is that arguments of h may be aliased, that is, appear several places as arguments to f or g . To handle the first problem, we observe that the subset relation is transitive. For example, consider

$$h(a_1, \dots, a_n) = f(a_1, \dots, a_i, g(a_{i+1}, \dots, a_j), a_{j+1}, \dots, a_n).$$

We wish to show that $(\forall i)(a_i \subseteq b_i) \rightarrow h(a_1, \dots, a_n) \subseteq h(b_1, \dots, b_n)$. Since g is monotonic, $(\forall i)(a_i \subseteq b_i) \rightarrow g(a_{i+1}, \dots, a_j) \subseteq g(b_{i+1}, \dots, b_j)$. This, coupled with the monotonicity of f , implies that h is monotonic. Aliasing does not cause any problems, as the requirement that all arguments obey the inclusion is still satisfied. ■

We will now explore some features of complementary, independent, and monotonic functions. We shall require not only that the domain of each function is a finite power set but also that the range be a finite power set as well. This is reasonable since the select, project, and join operators all result in power sets. Let f and g be complementary, independent, and monotonic functions whose domain is D and ranges are F and G (all finite power sets), respectively. Since f and g are complementary and independent, there is a one-to-one correspondence between G and the elements of any equivalence classes generated by f . Let G' be the equivalence class of f that contains the empty relation. Using the one-to-one correspondence, we can define $g' : D \rightarrow G'$ as $g'(d) = b$ where $g(b) = g(d)$ and $f(b) = \emptyset$. (Essentially, we have chosen from each equivalence class generated by g the element that maps to the emptyset by f .) Similarly, $f' : D \rightarrow F'$ is defined by $f'(d) = a$ where $f(a) = f(d)$ and $g(a) = \emptyset$. Let us define the coordination function $c : F' \times G' \rightarrow D$ as $c(f'(d), g'(d)) = d$. Since f and g are complementary, c is a function. Since f and g are independent, c is total. We will now explore some properties of c . First, $c(a, \emptyset) = a$ (and similarly, $c(\emptyset, b) = b$) since $a \in F'$ implies $f'(a) = a$ and $g'(a) = \emptyset$.

LEMMA. Let $f : F' \rightarrow F$ be a monotonic function. Then $f(\emptyset) = \emptyset$.

* The main property of finite power sets we use is that they are complete lattices [Birkhoff 67].

PROOF. Let $f(\emptyset) = a$, and let $f(d) = \emptyset$. (Both F and F' must contain the empty set.) Since $\emptyset \subseteq d$ and f is monotonic, $a \subseteq \emptyset$. Therefore, $a = \emptyset$. ■

LEMMA. F' is closed under containment.

PROOF. Let a_1 be a member of F' and a_2 be a subset of a_1 . Now $g(a_1) = \emptyset$ since $a_1 \in F'$. Since g is monotonic, $g(a_2) = \emptyset$. Therefore, a_2 is in F' . ■

COROLLARY. F' is closed under intersection.

LEMMA. Let $f : F' \rightarrow F$ be a monotonic function. If $d \in F'$ and $f(d) = \{e\}$, a singleton, then d is a singleton.

PROOF. Suppose not. Then there exist $d_1, d_2 \in d$ such that $d_1 \neq d_2$. By our previous lemma, both $\{d_1\}$ and $\{d_2\}$ are elements of F' . Let $f(\{d_1\}) = a_1$ and $f(\{d_2\}) = a_2$. We note that $a_1 \neq a_2$ since f is bijective on $F' \rightarrow F$. But the monotonicity of f implies that $a_1 \cup a_2 \subseteq \{e\}$. Therefore, $\{e\}$ is not a singleton. ■

LEMMA. Let $f : F' \rightarrow F$ be a monotonic function. If $d \in F'$ and $f(d) = a$, then $|d| \leq |a|$.

PROOF. Let S be the power set of d . Since F' is closed under containment, $S \subseteq F'$. For all s in S , $f(s)$ is in F and $f(s) \subseteq a$. Furthermore, since f is bijective on $F' \rightarrow F$, all the $f(s)$ are distinct. Then a has as many subsets as d , so a must be at least as big as d . ■

LEMMA. Let F and G be finite power sets. Then F' (and also G') is a finite power set.

PROOF. Since F and G are finite power sets, their cardinalities are powers of two. Let $|F| = 2^m$ and $|G| = 2^n$. Then F (G) contains exactly m (n) singletons. Since there is a one-to-one correspondence between D and $F \times G$, $|D| = 2^{m+n}$. Because D is a finite power set, so D contains exactly $m+n$ singletons. Let S_F (S_G) be the singletons in F' (G') that map into singletons in F (G). We note that $|S_F| = m$ and $|S_G| = n$ since each singleton of F is mapped into by a unique singleton of F' (and consequently S_F). Since $f : F' \rightarrow F$ is a bijection, $|F'| = 2^m$. Now, suppose that F' is not the power set generated by S_F . Since $|S_F| = m$, the power set of S_F is of size 2^m . As F' is the same size as the power set generated by S_F and they are unequal, there must be some element a in F' that is not in the power set of S_F . That set a in F' is then not the union of some singletons S_F . Then there is some element $a_0 \in a$ not in S_F . Then $\{a_0\}$ is a singleton that is not in S_F . Then F' has more than m singletons. Let us now show that $F' \cap G' = \{\emptyset\}$. Everything in F' is mapped to \emptyset by g , while the only set in G' mapped to \emptyset by g is \emptyset . Therefore, the singletons in F' and G' are disjoint. But F' has more than m singletons while G'

has at least n singletons. This contradicts the fact that D has precisely $m + n$ singletons. ■

LEMMA. Let the domain $F, F', G,$ and G' be finite power sets, and let $f : F' \rightarrow F$ be a monotonic function. If $d \in F'$ and $f(d) = a$, then $|d| = |a|$.

PROOF. We have already shown that $|d| \leq |a|$. Let $s_i = \{s \in F' \mid |s| = i\}$. Let $t_i = \{t \in F \mid |t| = i\}$. Since F and F' are power sets of size 2^n , $|s_i| = |t_i| = \binom{n}{i}$. By induction on $|d|$ we will show that $|d| = |a|$. For $|d| = 1$, s_1 is the set of singletons in F' . Since $|d| \leq |a|$, only singletons (elements of s_1) may map into the t_1 . But since $|s_1| = |t_1| = \binom{n}{1}$, all elements of s_1 must map into elements of t_1 . For the induction step, assume that all elements of s_i map into elements of t_i for $1 \leq i < j \leq n$. We will show that all elements of s_j map into elements of t_j . Since $|d| \leq |a|$, the elements of F' that map into elements of t_j must be elements of s_i for $1 \leq i \leq j$. But by the induction hypothesis, none of these can be elements of s_i for $1 \leq i < j$. Therefore only elements of s_j can map into elements of t_j . But $|s_j| = |t_j| = \binom{n}{j}$. Therefore all elements of s_j map into elements of t_j . ■

THEOREM. Let the domain D be a finite power set. Then F' and F are isomorphic (preserving monotonicity) under f .

PROOF. The preceding lemma ($|d| = |a|$) showed that the singletons of F and D are in one-to-one correspondence. Let $d = \{a_1, \dots, a_k\}$. We will show that $f(\{a_1, \dots, a_k\}) = \{f(a_1), \dots, f(a_k)\}$. Suppose not. Then there is some $f(a_i)$ ($1 \leq i \leq k$), say $f(a_1)$, not in $f(d)$. (From $|d| = |f(d)|$, we know that some of the a_i ($1 \leq i \leq k$) are missing and there are other singletons added.) By definition, $\{a_1\} \subseteq d$. But since f is monotonic, this implies $f(\{a_1\}) \subseteq f(d)$. ■

COROLLARY. f' is monotonic.

Coordinatization

LEMMA. Let the domain D be a finite power set, and let f and g be monotonic, independent, and complementary. Then all the singletons of D are members of F' or G' .

PROOF. Let 2^m (2^n) be the size of F (G). Then $|F'| = 2^m$ and $|G'| = 2^n$. Since F' (G') is a power set, there are m (n) singletons in F' (G'). Also, $|D| = 2^{m+n}$. Since D is a power set, there are $m + n$ singletons in D . We know that $F' \cap G' = \{\emptyset\}$. Then all of the singletons of D are members of F' or G' . ■

COROLLARY. For all $d \in D$, there exist $a \in F'$ and $b \in G'$ such that $d = a \cup b$.

THEOREM. Let the domain D be a finite power set. For c as the coordinatization function defined above, $c(a, b) = a \cup b$. (That is, f and g are monotonic, independent, and complementary.)

PROOF. We observe that the theorem holds of $c(a, \emptyset) = a$ and $c(\emptyset, b) = b$. We measure a counterexample $c(a, b) = a' \cup b'$ where $a \neq a'$ or $b \neq b'$ and $a' \in F'$ and $b' \in G'$ by the sum of the cardinalities of a and b ($|a| + |b|$). We perform an induction on this measure. Assume that $c(a, b) = a \cup b$ for all a and b such that $|a| + |b| < n$. Let $c(a, b) = a' \cup b'$ with $a \neq a'$ or $b \neq b'$ and $|a| + |b| = n$. It is not possible that $a' \subseteq a$ and $b' \subseteq b$. (Otherwise either $a' \cup b' = a \cup b$ —assumed false—or $|a' \cup b'| < |a \cup b|$, which by our induction hypothesis implies $c(a', b') = a' \cup b'$.) Without loss of generality, assume that $a' - a \neq \emptyset$. Let $e \in a' - a$. Then $\{e\} \subseteq a' \cup b'$, so $f'(\{e\}) \subseteq f'(a' \cup b') = a$. Since $e \in a'$, $\{e\} \in F'$ and $f'(\{e\}) = \{e\}$. Since $e \in a' - a$, $f'e \notin a$, a contradiction. ■

Let us consider the consequence of the preceding theorem and lemmata. Let the B be the basis set of the domain D . (That is D is the power set of B .) Also, let B_f be the basis set of F' and B_g be the basis set of G' . Then $B = B_f \cup B_g$.

THEOREM. Let the domain D be a finite power set. Coordinatization preserves monotonicity. That is, let f and g be monotonic, complementary, and independent functions with domain D and ranges F and G respectively (all finite power sets), and let h be a function with domain D , and let h' be the coordinatization of h over f and g [that is, $h'(f(d), g(d)) = h(d)$]. Then h is monotonic iff h' is.

PROOF. “If.” We note that h is the composition of $f, g,$ and h' . Therefore, if h' is monotonic, then h is also.

“Only if.” We note that h' is the composition of h and c . If h is monotonic, then h' is also since c is monotonic. ■

Uniqueness of Independent, Complementary Mappings

Our question now becomes when does a mapping have a unique (up to equivalence) monotonic, complementary, independent mapping. For domains of finite sets, such mappings are unique.

THEOREM. Let $f, g,$ and h be monotonic mappings on a domain D , a finite power set, such that the ranges are all finite power sets, and f and g are independent and complementary, as are f and h . Then g and h are equivalent.

PROOF. Let f' , g' , and h' be the coordinatizations of f , g , and h , respectively, over f and g . Let F be the range of f and G be the range of g . Assume that g and h are not equivalent (and consequently, g' and h'). Then there exists some $a \in F$ and $b_0 \in G$ such that $g'(a, b_0) = g'(\emptyset, b_0)$ but $h'(a, b_0) \neq h'(\emptyset, b_0)$. Since coordinatization preserves monotonicity, $h'(\emptyset, b_0) \subset h'(a, b_0)$. Choose b_1 such that $h'(a, b_1) = h'(\emptyset, b_0)$. (Such a b_1 must exist since f' and h' are independent.) Since f' and h' are complementary, $h'(a, b_1) \neq h'(\emptyset, b_1)$. Since coordinatization preserves monotonicity, $h'(\emptyset, b_1) \subset h'(a, b_1)$. Now we have $h'(\emptyset, b_1) \subset h'(a, b_1) = h'(\emptyset, b_0) \subset h'(a, b_0)$. We can choose b_{i+1} such that $h'(a, b_{i+1}) = h'(\emptyset, b_i)$. This defines an infinite sequence of sets, each of which is a proper subset of the previous one. This is not possible when the domain is finite sets. Thus, we have arrived at a contradiction. ■

Conclusion

We have considered the relationship between independent and complementary mappings. We have shown that on databases on finite domains, when mappings are monotonic, for each mapping there is at most one other mapping that is independent and complementary to it. The domains and ranges are all finite power sets. An intersection mapping intersects each domain element with a fixed set (called the intersection set) to produce the result. Given a pair of mappings that are monotonic, independent, and complementary, the two mappings are equivalent to intersection mappings where the intersection sets are a partition of the generator of the power set.

Acknowledgements

The concepts and most of the work is by Arthur Keller. Jeff Ullman suggested key steps of some of the proofs. Gio Wiederhold provided encouragement and support. Shel Finkelstein noted the importance of monotonicity to database views. Christos Papadimitriou, Al Aho, Don Knuth, Jim Davidson, Richard Beigel, and Ken Clarkson gave advice or commented on drafts of the manuscript.

Bibliography

- [Bancilhon 79] F. Bancilhon, "Supporting View Updates in Relational Data Bases," in *Data Base Architecture*, Bracci and Nijssen, eds., North Holland, June 1979.
- [Bancilhon 81] F. Bancilhon and N. Spyrtos, "Update Semantics and Relational Views," *ACM Trans. on Database Systems*, 6:4, December 1981.
- [Birkhoff 67] Garrett Birkhoff, *Lattice Theory*, American Mathematical Society, Providence, RI, Colloquium Publications, Volume 25, 1967.
- [Carlson 79] C. Robert Carlson and Adarsh K. Arora, "The Updatability of Relational Views Based on Functional Dependencies," *Third International Computer Software and Applications Conference*, IEEE Computer Society, Chicago, IL, November 1979.
- [Davidson 81] Jim Davidson and S. Jerrold Kaplan, "Natural Language Access to Databases: Interpretation of Update Requests," *Proc. 7th Int. Joint Conf. on Artificial Intelligence*, Vancouver, B.C., August 1981.
- [Dayal 78] U. Dayal and P. A. Bernstein, "On the Updatability of Relational Views," *Proc. Fourth VLDB Conf.*, IEEE Computer Society, Berlin, West Germany, October 1978.
- [Dayal 79] Umeshwar Dayal, *Schema-Mapping Problems in Database systems*, Aiken Computation Laboratory, Harvard University, TR-11-79, Ph.D. dissertation, August 1979.
- [Dayal 82] U. Dayal and P. A. Bernstein, "On the Correct Translation of Update Operations on Relational Views," *ACM Trans. on Database Systems*, 7:3, September 1982.
- [Furtado 79] A. L. Furtado, K. C. Sevcik, and C. S. DOS Santos, "Permitting Updates Through Views of Data Bases," *Inform. Systems*, 4, 4, Pergamon Press, Great Britain, 1979.
- [Halmos 60] Paul R. Halmos, *Naive Set Theory*, Springer-Verlag, New York, 1960.
- [Kaplan 81] S. Jerrold Kaplan and Jim Davidson, "Interpreting Natural Language Database Updates," *Proc. 19th Annual Meeting of the Association for Computational Linguistics*, Stanford, California, June 1981.
- [Keller 82] Arthur M. Keller, "Updates to Relational Databases Through Views Involving Joins," in *Improving Database Usability and Responsiveness*, Peter Scheuermann, ed., Academic Press, New York, 1982.
- [Rissanen 77] Jorma Rissanen, "Independent Components of Relations," *ACM Trans. on Database Systems*, 2:4, December 1977.
- [Stonebraker 75] Michael Stonebraker, "Implementation of Integrity Constraints and Views by Query Modification," *Proc. of the 1975 SIGMOD Conference*, ACM SIGMOD, San Jose, June 1975.
- [Ullman 82] Jeffrey D. Ullman, *Principles of Database Systems*, Computer Science Press, Potomac, MD, second edition, 1982.